Optimal serving schedules for multiple queues with size-independent service times

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Yuhang Liu

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Zizhuo Wang

May, 2013

Acknowledgements

This research project would have not been possible without the support of many people. I would like to express my sincere gratitude to everyone who contributed to this thesis.

Foremost, I would like to express my sincere gratitude to my advisor Prof. Zizhuo Wang for his continuous support of my MS study and research, for his patience, motivation, enthusiasm, and immense knowledge. His insightful guidance helped me a lot in my research and thesis writing. I also would like to thank the rest of my thesis committee: Prof. John Gunnar Carlsson and Prof. Jasmine Foo for joining my defense.

Besides, I would like to thank my study group member Ernesto Nunes for his valuable suggestions in my research and working with me together on some problems. I would never forget those days. My sincere thanks also go to Qiushi Li for his kind and helpful discussion with me on many interesting topics.

Finally, I would like to express my gratitude to all my friends: Jin Lin, Yi Wan, Meng Shi, Jose Garcia, Ajay Vadakkepatt for the encouragement, and the fun we had together in the past years.

Abstract

We consider a service system with two Poisson arrival queues. There is a single server that chooses which queue to serve at each moment. Once a queue is served, all the customers are served within a fixed time. This model is useful in studying airport shuttling or certain online computing systems. In this thesis, we first establish a Markov Decision Process (MDP) model for this problem and study its structures. We then propose a simple yet optimal state-independent policy for this problem which is not only easy to implement, but also performs very well. If the service time of both queues equals to one unit of time, we prove that the optimal state-independent policy has the following structure: serve the queue with the smaller arrival rate once followed by serving the other queue k times, and we obtain an explicit formula to capture k. We conduct numerical tests for our policy and it performs very well. We also extend our discussions to a more general case in which the service time of the queues can be any integer. We also obtain the optimal the optimal state-independent policies in that case.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Finding efficient and practical service plans for multi-queue systems has always been an important question both for researchers and practitioners. In this thesis, we consider a special type of multi-queue system that has the following features: 1) the arrivals in each queue are Poisson processes; 2) there is only one server; 3) once a queue is chosen to be served, all the customers in that queue will be served in a fixed amount of time. The decision in such a system is to decide at each time, which queue should be served and the overall objective is to minimize the total expected delays of the customers.

The above model covers a wide range of problems that may occur in practice. For example, in an airport transportation system, a shuttle bus picks up passengers from multiple locations (e.g., different rental car locations or different train stops) and drops them off at the terminal. Due to the route constraints, the shuttle bus can only go to one location at a time (i.e., a cyclic route is not permissible). Customers arrive at each location with a certain rate and the service provider has to decide its service schedule (the schedule could be either state dependent or independent) to minimize the total delays of all customers. In such problems, assuming that the capacity of the shuttle is sufficiently large, the bus can pick up all the passengers waiting in the location that it decides to serve and the service time depends little on the job size (it mainly depends on the distance of the location). Thus, this problem can be captured well by our model. For another example, in an online computing service system, a server provides computational services for several types of customers. In some applications, the service time mainly depends on the setup time (software initializations and warm up, etc.) and depends little on the job size. In this case, again, the decision for the server is to decide which type of customer to serve at each time period, with an overall objective of minimizing the total delays of the jobs.

In such problems, the difficulty is the need to consider both the system state (the current customers waiting in each queue) and the forthcoming customers. Since different queues may have different arrival rate, different attention needs to be taken to each of them. A natural intuition is to serve the queue with higher arrival rate more frequently, but how frequent should we serve? If a certain queue is served too frequently while another one too infrequently, even the arrival rate of the infrequently-served queue is smaller, the aggregate waiting time over a period may still be dominant in the cost function. In this thesis, we establish models to capture this tradeoff and propose optimal policies for this problem.

We start with the problem with only two queues. We formulate this problem as a Markov Decision Process (MDP). Then we study some basic properties of this MDP problem to illustrate some basic behaviors of the optimal decision as well as the value function. We also discuss how to solve this MDP. See Chapter 2 for these discussions. Although the MDP formula is a precise description of the actual problem and the policy obtained therefore is optimal, the optimal policy has the feature of being state-dependent. However, in practice, such a feature might be undesirable due to the limit knowledge of the state of the system and the implementation constraints. To solve this problem, in Chapter 3, we propose a simple state-independent policy that is independent of the current state. Among the state-independent policies, we prove that the optimal one must consist of cycles of a certain fixed schedule, and each cycle consists of serving the slow-arriving queue once followed by serving the fast-arriving queue multiple (k) times. We then give an explicit formula for the optimal k. In particular, we show that when the discount factor across periods approaches 1, the optimal k is approximately $\sqrt{2r} - 1$, where r is the ratio of the arrival rate between the fast-arriving queue and the slow one. This neat result is significant as it provides a quantified optimal strategy for the service providers of how often should the slow-arriving queue be served. We implement this strategy in our numerical experiments and show that it performs very well.

Finally, in Chapter 4, we extend our model to the cases in which queues have heterogeneous service times. We obtain similar results for that case.

Before we proceed to the models, we review some related literature in the following.

1.1 Literature review

Our work is related to the batched service systems in queuing theory. For a comprehensive review of the basics of batched service system, we refer the readers to the books by Chaudhry [2], Cooper [3] and Gross et al [9].

In particular, our work is related to a vast literature in designing systems where a shuttle runs between two or several terminals. In those problems, a shuttle runs between two terminals and sends customers from one terminal to the other. The studies of those problems can be classified by whether the shuttle is finitely capacitated [4, 5] or there is no capacity limit [10, 13, 14]; whether the control can be exerted in both terminals [5, 4, 13, 12] or can only be exerted at a single terminal [10, 14, 11]. The cost usually consists of the operating costs, that is, costs are charged for each transportation based on a fixed cost and a per passenger costs, and the waiting costs of the customers. In this literature, the control policy is usually of the form of a threshold policy, that is, if the number of passengers waiting at one terminal (or either of them) is greater than a certain threshold, then one should dispatch immediately. In our work, we assume there are two queues, and the service provider can serve one of them at each time. If one views this in the shuttle context, instead of sending passengers between two terminals, we capture the situations that a shuttle picks up (or sends) passengers from (or to) two terminals and send them to (or from) a common location. And our primary decision is the service schedule between these two queues (whereas they don't have this problem since the service is always run between these two terminals). Therefore, although the background shares some similarity, our model is different from this line of research.

Another research area that is related to our work is the area of vehicle routing problems. We refer the readers to Laporte [8] for a comprehensive review of this literature. The main difference between our work and the vehicle routing problems is that we don't design the route. Instead, we design the service schedule to several locations. And there is no capacity or time windows for our service. Moreover, our load is stochastic, and our criterion is the expected waiting time.

In some sense, our model also relates to the problem of traffic controls. In those problems, there are also two (or more) arrival queues, and one has to determine when to serve each of them. A through study of this problem is provided in [6] and literature thereafter. Although we share some similarities in terms of the state space, the model we use is quite different in that we assume all the customers can be served in a fixed amount of time, regardless of the backup, whereas in the traffic control literature, usually a departure rate is used. This difference distinguishes our analysis and control policy from that of the traffic control literature.

Chapter 2

MDP Model and Analysis

We consider a system of two independent queues without capacity limits. Customers arrive in each of these two queues according to Poisson processes, with rates λ_1 and λ_2 respectively. One distinguishing feature of our model is that at each time the server is idle, it could choose one of the two queues to serve. And once a queue is chosen to be served, all of its current customers will be served (and depart the system) within a deterministic time.[1] This model characterizes situations in which the service time is largely determined by the setup time but not how much service it needs to provide. For example, in an airport shuttle service system, a shuttle needs to pick up passengers from two locations and send them to the terminal. In such cases, as long as the capacity of the shuttle is large enough, it is reasonable to assume that at each time, all the passengers from one location can all be served. Moreover, the service time of each cycle mainly depends on the distance from the service location to the terminal and depends little on how many passengers it serves. Similarly, in an online computing service system, two types of licensed computing service are provided. For certain type of services, the service time is dominated by the set-up time (e.g., which software to use, and initializations), rather than the amount of jobs. In such cases, again, the service time can be assumed to be constant and all the customers of the same type can be served all at once in a fixed amount of time. In the following discussion, we first discuss the case in which the service time for both queues are the same. Without loss of generality, we assume that

[1] However, the customers that arrive after the service has started will not be served during this service cycle.

the service time equals to 1.

Throughout this thesis, we use Q_i, $i = 1, 2$ to denote the ith queue. Due to the assumption of our problem, we can consider this problem in a number of discrete time periods. The decisions in our model is to decide a queue to serve at each time period. And the objective is to determine a service rule such that the expected discounted total waiting time of all the customers is minimized. To mathematically capture this objective function, we adopt a Markov Decision Process (MDP) approach. We use $s = (x, y)$ to denote the state when there are x people waiting in the first queue and y people waiting in the second queue. We use $V(x, y)$ to denote the optimal expected discounted waiting time onward (the cost function) when the state is (x, y). At each state, the action set $\mathbf{A} = \{\mathbf{A_1}, \mathbf{A_2}\}$ contains two actions: $\mathbf{A_1}$ which means to serve Q_1 and $\mathbf{A_2}$ which means to serve Q_2. We use Z_i to denote the amount of customers that arrive at Q_i during one service period. By our assumption, Z_i follows a Poisson distribution with parameter λ_i. We use $P_a(s, s')$ to denote the probability that by taking action $a \in \mathbf{A}$ the state changes from s to s' and $R_a(s, s')$ to denote the immediate reward after taking action a and when the state changes from s to s'.

The Bellman equation for this MDP problem can be written as

$$V(x, y) = \frac{\lambda_1 + \lambda_2}{2} + \min\{\gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y, \gamma\mathbb{E}[V(Z_1 + x, Z_2)] + x\}. \tag{2.1}$$

In (2.1), the first term is the expected waiting time for the arriving customers during the current period. Note that if the arrivals are Poisson processes, given the number of arrivals in one period, the exact arrival time during this period is uniformly distributed. Therefore, the expected waiting time of the new arrivals are $\lambda_i/2$ for each queue i. The second term in (2.1) corresponds to the two choices available at the current period, i.e., to serve either the first queue or the second queue. When the first queue is served, the state will change to $(Z_1, Z_2 + y)$ and the expected waiting time occurred in this time period for the customers currently in the queue is y. Similarly, when the second queue is served, the state will change to $(Z_1 + x, Z_2)$ and the expected waiting time occurred in this time period for the customers currently in the queue is x. γ is the discount factor for the later periods.

2.1 Solution procedure

The MDP defined in (2.1) is an infinite state problem. In order to solve this problem, it is desirable to reduce it to a finite state problem and then we can apply standard techniques to solve it. In the following, we describe the method we employ to solve this problem.

To transform an infinite state space MDP to a finite state space one, a natural step is to truncate the state space. In this particular case, it is easy to note that the probability a certain state (i.e., the number of people waiting in each of the queues) taking large values is small. The reason is two-fold. First, the chance of getting a large arrival in any period is small due to the property of Poisson random variables. Second, it is obvious suboptimal not to serve a very long queue given the other one is much shorter. Therefore, the chance that the queue length will be large at any time period is very small. Thus, it is reasonable to consider truncating the state space at a large number. In our solution approach, we truncate each queue i at length $\bar{\lambda}_i = \lambda_i + 10\sqrt{\lambda_i}$ (note that if $\lambda > 1$, then $P(Z_i \geq \bar{\lambda}_i) < 10^{-7}$). That is, we use the following Bellman equation as the approximation for the original MDP (2.1).

$$\begin{aligned} V(x,y) \quad = \quad & \frac{\lambda_1+\lambda_2}{2} + \min\{\gamma\mathbb{E}[V(\min(Z_1,\bar{\lambda}_1),\min(Z_2+y,\bar{\lambda}_2))]+y, \\ & \gamma\mathbb{E}[V(\min(Z_1+x,\bar{\lambda}_1),\min(Z_2,\bar{\lambda}_2))]+x\}. \end{aligned} \tag{2.2}$$

To solve (2.2), we use the standard value iterations method. We start with $V(x,y)=0$, and iteratively update $V(x,y)$ using the relation in (2.2) until the change for the value function is small enough. By [1], this procedure converges to the true optimal values for (2.2).

2.2 Properties of the optimal policy and the value function

In the following, we study the properties of the optimal policy and the value function in (2.1). We first show that

Theorem 1. *$V(x,y)$ is an increasing concave function of (x,y).*

Proof. Define $V_0(x,y)=0$ and

$$V_{k+1}(x,y) = \frac{\lambda_1+\lambda_2}{2} + \min\{\gamma\mathbb{E}[V_k(Z_1,Z_2+y)]+y, \gamma\mathbb{E}[V_k(Z_1+x,Z_2)]+x\} \tag{2.3}$$

for all $k \geq 0$. By Proposition 1.2.1 of [1], since $V_0(x,y)$ is bounded, we have that

$$\lim_{k\to\infty} V_{k+1}(x,y) = V(x,y). \tag{2.4}$$

Now we claim that each V_k is increasing and concave. We use induction. For $k=0$, the claim is trivial. Suppose the claim holds for $k=l$, then for $k=l+1$, we first observe that both terms in the right hand side of (2.3) are increasing in (x,y). Furthermore, since $V_k(x,y)$ is concave by induction assumption, both $\mathbb{E}[V_k(Z_1, Z_2+y)]$ and $\mathbb{E}[V_k(Z_1+x, Z_2)]$ are concave functions in y and x respectively. Thus, both $\gamma\mathbb{E}[V_k(Z_1, Z_2+y)]+y$ and $\gamma\mathbb{E}[V_k(Z_1+x, Z_2)]+x$ are concave functions in (x,y) and so is their minimum. Therefore V_{k+1} is also increasing and concave. By induction principle, the claim holds. And finally by taking $k\to\infty$ in (2.4), the theorem is proved. □

Remark 1. *One natural extension of the baseline problem in (2.1) is to allow the state variables to take any positive real numbers, i.e., we do not restrict x and y to be integers. Correspondingly, we can use a continuous Poisson distribution to model the arrival process which has a density function of*

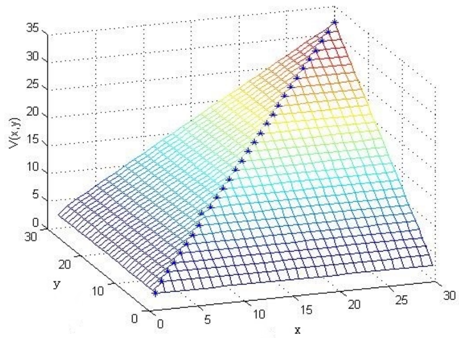
$$f_\lambda(x) = \frac{\lambda^x e^{-\lambda}}{\Gamma(x+1)} \quad \text{for } x \geq 0.$$

Note that this does not change the dynamic programming formula (2.1). And Theorem 1 still holds.

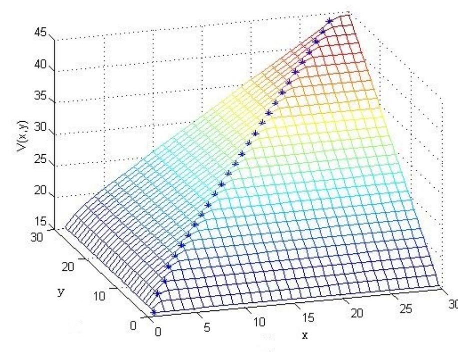
In Figure 2.1, we show some numerical results of the function $V(x,y)$. In Figure 2.1, we use $\gamma = 0.8$, $\lambda_1 = 1$ and $\lambda_2 = 1, 3, 10$ and 15 respectively. We can see that $V(x,y)$ is indeed an increasing and concave function, which is consistent with Theorem 1. Moreover, we can see that the surface of $V(x,y)$ is the intersection of two surfaces. We will show this fact in the next theorem.

In the following, for the ease of notation, we denote $g_1(x) = \mathbb{E}[V(Z_1+x, Z_2)]$, $g_2(y) = \mathbb{E}[V(Z_1, Z_2+y)]$ and $\frac{\lambda_1+\lambda_2}{2} = \lambda$. We next study the property of the optimal decisions at each state in the MDP. We have the following theorem:

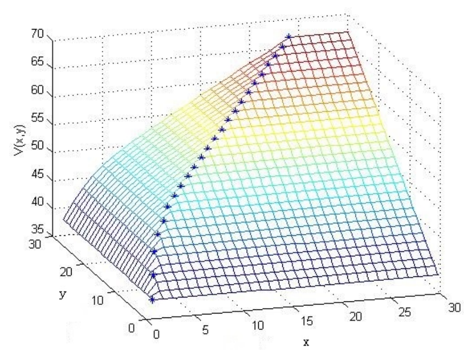
Theorem 2. *$\mathbb{R}^2_+$ can be divided by an increasing function $y = W(x)$, such that, for $y > W(x)$, the optimal action is $\mathbf{A_2}$, and for $y < W(x)$, the optimal action is $\mathbf{A_1}$. When $y = W(x)$, there is no difference between using $\mathbf{A_1}$ or $\mathbf{A_2}$.*



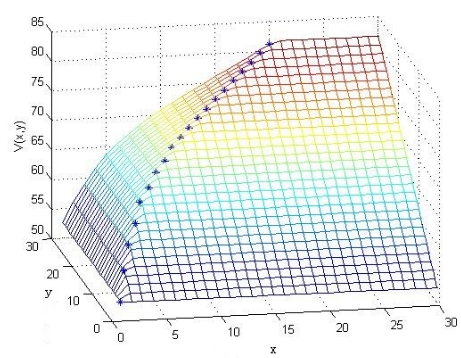
(a) $\lambda_1 = 1$, $\lambda_2 = 1$



(b) $\lambda_1 = 1$, $\lambda_2 = 3$



(c) $\lambda_1 = 1$, $\lambda_2 = 10$



(d) $\lambda_1 = 1$, $\lambda_2 = 15$

Figure 2.1: Numerical results of $V(x, y)$

Proof. For each given x^*, by the intermediate value theorem and the monotonicity of V, there must exists a unique y^* such that

$$\gamma\mathbb{E}[V(Z_1 + x^*, Z_2)] + x^* = \gamma\mathbb{E}[V(Z_1, Z_2 + y^*)] + y^*.$$

We denote this relationship by $y = W(x)$. By the monotonicity of $V(x, y)$, we have

$$\begin{aligned} V(x^*, y) &= \lambda + \min\{\gamma\mathbb{E}[V(Z_1 + x^*, Z_2)] + x^*, \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y\} \\ &= \lambda + \gamma\mathbb{E}[V(Z_1 + x^*, Z_2)] + x^* \end{aligned}$$

for $y > y^*$ and it is optimal to choose to serve Q_2. Similarly, when $y < y^*$, we have

$$\begin{aligned} V(x^*, y) &= \lambda + \min\{\gamma\mathbb{E}[V(Z_1 + x^*, Z_2)] + x^*, \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y\} \\ &= \lambda + \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y \end{aligned}$$

and it is optimal to choose to serve Q_1. Therefore, the theorem is proved. $\square$

Following from the proof of Theorem 2, we immediately have the following corollary:

Corollary 1. *For any $y \in \mathbb{R}_+$,*

$$\lim_{x \to +\infty} V(x, y) = \lambda + \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y$$

and for any $x \in \mathbb{R}_+$,

$$\lim_{y \to +\infty} V(x, y) = \lambda + \gamma\mathbb{E}[V(Z_1 + x, Z_2)] + x.$$

In the following, we define $V_1^*(x) = \lim_{y \to +\infty} V(x, y)$ and $V_2^*(y) = \lim_{x \to +\infty} V(x, y)$. By Corollary 1, they are well defined and moreover, by Theorem 1, both $V_1^*(x)$ and $V_2^*(y)$ are concave.

We give a further analysis of the function $W(x)$ in the next theorem.

Theorem 3. *1. If $\lambda_1 = \lambda_2$, then $W(x) = x$;*

2. $x + \gamma(V(0, 0) - V_1^*(\lambda_1)) \le W(x) \le x + \gamma(V_2^*(\lambda_2) - V(0, 0))$;

Proof. By definition, $y = W(x)$ is the unique solution to

$$\gamma\mathbb{E}[V(Z_1 + x, Z_2)] + x = \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y. \tag{2.5}$$

When $\lambda_1 = \lambda_2$, it is easy to see that the two queues are symmetric, therefore $V(x, y) = V(y, x)$ and $y = x$ will be the unique solution to (2.5). Therefore, $W(x) = x$ in this case.

For the second part, we first show the second inequality, note that

$$\begin{aligned} W(x) &= x + \gamma\mathbb{E}[V(Z_1 + x, Z_2) - V(Z_1, Z_2 + W(x))] \\ &\leq x + \gamma(V(\lambda_1 + x, \lambda_2) - V(0, 0)) \\ &\leq x + \gamma(V_2^*(\lambda_2) - V(0, 0)). \end{aligned}$$

Here the first inequality is because $V(x, y)$ is an increasing and concave function of (x, y) and the Jensen's inequality. The second inequality is due to the monotonicity of $V(x, y)$ and our definition of $V_2^*(\cdot)$. Similarly, we have

$$\begin{aligned} W(x) &= x + \gamma\mathbb{E}[V(Z_1 + x, Z_2) - V(Z_1, Z_2 + W(x))] \\ &\geq x + \gamma(V(0, 0) - V(\lambda_1, \lambda_2 + W(x))) \\ &\geq x + \gamma(V(0, 0) - V_1^*(\lambda_1)). \end{aligned}$$

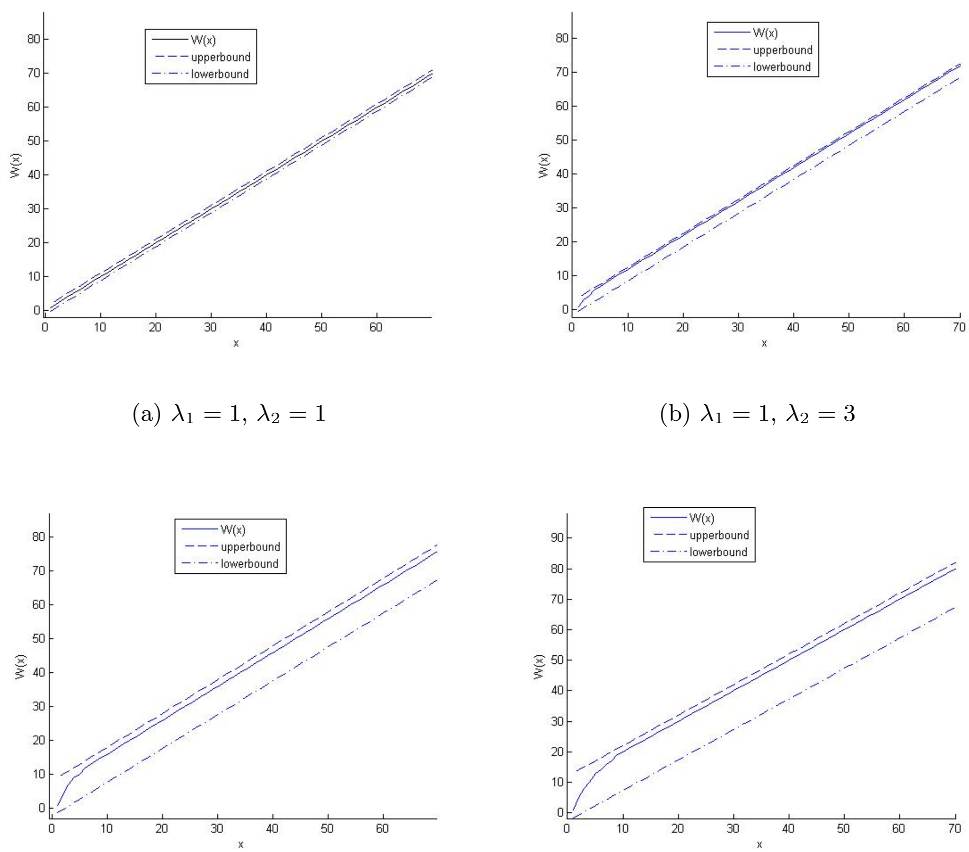
Therefore the theorem holds. □

In Figure 2.2, we show some examples of $W(x)$ and the upper and lower bounds given in Theorem 3. Notice that the upper bounds are quite tight. We also find than when $\lambda_2 > \lambda_1$, $W(x)$ is always greater than x. If this is true, it will be a tighter lower bound for $W(x)$ than that is given in Theorem 3 (similarly we find that when $\lambda_1 > \lambda_2$, we always have $W(x) < x$). This property of $W(x)$ is intuitively true. When $\lambda_2 > \lambda_1$, there is a higher chance that Q_2 will be served in the next service period. Therefore, to keep the waiting time in Q_1 small, it should be better to serve Q_1 at state $s = (x, x)$. Despite this intuitive explanation, we were unable to prove it and will leave it as one of the future work.

In Figure 2.3, we put several curves $y = W(x)$ together to show that as λ_2 increases, the curve tends to be more concave. And also, all the curves tends to be linear with slope 1 as x increases. We explain this phenomenon in Theorem 4.

Theorem 4.

$$\lim_{x \to +\infty} \{W(x) - x\} = \gamma(\mathbb{E}_{Z_2} V_2^*(Z_2) - \mathbb{E}_{Z_1} V_1^*(Z_1))$$



(a) $\lambda_1 = 1$, $\lambda_2 = 1$

(b) $\lambda_1 = 1$, $\lambda_2 = 3$

(c) $\lambda_1 = 1$, $\lambda_2 = 10$

(d) $\lambda_1 = 1$, $\lambda_2 = 15$

Figure 2.2: Lower and upper bound of $W(x)$

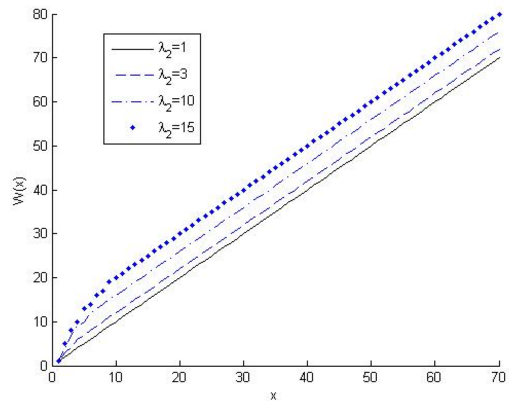


Figure 2.3: Asymptotic behavior of $W(x)$

Proof. We prove by showing the result for any subsequence x_n such that $x_n > x_{n-1}$, $\lim_{n\to\infty} x_n = \infty$. If this holds, then the whole sequence must also converge to the same limit. For any subsequence $\{x_n\}_{n=1}^{\infty}$, we have

$$\lim_{n\to+\infty} \mathbb{E}[V(Z_1 + x_n, Z_2)] \;=\; \mathbb{E}_{Z_2} V_2^*(Z_2). \tag{2.6}$$

This can be shown by using the monotone convergence theorem [7] and noting that $V(Z_1 + x_n, Z_2)$ is monotonically increasing in n. Therefore, we can change the order of limit and expectation and (2.6) holds. Apply this to $\lim_{x\to+\infty}\{W(x) - x\} = \lim_{x\to+\infty}\{\gamma\mathbb{E}[V(Z_1 + x, Z_2) - V(Z_1, Z_2 + W(x))]\}$. We have the desired result. □

Next we discuss some properties of the function $V(x, y)$ itself. In particular, we obtain two upper bounds of $V(x, y)$ in Theorem 5 and 6.

Theorem 5.

$$V(x, y) \;\leq\; \lambda + \min\{\gamma V_1^*(\lambda_1) + y, \gamma V_2^*(\lambda_2) + x\}$$

Proof.

$$\begin{aligned} V(x, y) &= \lambda + \min\{\gamma\mathbb{E}[V(Z_1 + x, Z_2)] + x, \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y\} \\ &\leq \lambda + \min\{\gamma V(\lambda_1 + x, \lambda_2) + x, \gamma V(\lambda_1, \lambda_2 + y) + y\} \\ &\leq \lambda + \min\{\gamma V_2^*(\lambda_2) + x, \gamma V_1^*(\lambda_1) + y\} \end{aligned}$$

where the first inequality is due to Jensen's inequality. □

Corollary 2.

$$\begin{aligned} V(x,y) &\leq \lambda + \min\{x,y\} + \gamma \max\{V_2^*(\lambda_2), V_1^*(\lambda_1)\} \\ &= \min\{x,y\} + C \end{aligned}$$

where $C = \lambda + \gamma \max\{V_2^*(\lambda_2), V_1^*(\lambda_1)\}$ *is a constant.*

.

Theorem 6.

$$V(x,y) \leq \frac{1}{(1-\gamma)^2} \min\{\lambda_1\gamma + (1-\gamma)(\lambda + x), \lambda_2\gamma + (1-\gamma)(\lambda + y)\}$$

Proof. First, we note that by definition:

$$\begin{aligned} V(x,y) &= \lambda + \min\{\gamma\mathbb{E}[V(Z_1 + x, Z_2)] + x, \gamma\mathbb{E}[V(Z_1, Z_2 + y)] + y\} \\ &\leq \lambda + \gamma\mathbb{E}_{Z_1,Z_2}[V(Z_1 + x, Z_2)] + x. \end{aligned} \tag{2.7}$$

Repeatedly apply (2.7), we have

$$\begin{aligned} V(x,y) &\leq \lambda + \gamma\mathbb{E}_{Z_1,Z_2}[V(Z_1 + x, Z_2)] + x \\ &\leq \lambda + \gamma\mathbb{E}_{Z_1,Z_2}[\lambda + \gamma\mathbb{E}_{Z_3,Z_4}[V^*(Z_1 + Z_3 + x, Z_2 + Z_4) + x + Z_1]] + x \\ &= \lambda + \gamma\lambda + x + \gamma x + \gamma\mathbb{E}[Z_1] + \gamma^2\mathbb{E}_{Z_1,Z_2,Z_3,Z_4}[V(Z_1 + Z_3 + x, Z_2 + Z_4)] \\ &\leq \lambda + \gamma\lambda + \gamma^2\lambda + x + \gamma x + \gamma^2 x + \gamma\mathbb{E}[Z_1] + \gamma^2\mathbb{E}_{Z_1,Z_2,Z_3,Z_4}[Z_1 + Z_3] + \\ &\quad \gamma^3\mathbb{E}_{Z_1,Z_2,Z_3,Z_4,Z_5,Z_6}[V(Z_1 + Z_3 + Z_5 + x, Z_2 + Z_4 + Z_6)] \end{aligned}$$

and we know that $\mathbb{E}[Z_1] = \lambda_1$ and $\mathbb{E}[Z_1 + Z_3] = 2\lambda_1$ and so on so forth. Thus, we have:

$$\begin{aligned} V(x,y) &\leq \lambda(1 + \gamma + \gamma^2 + ... + \gamma^{n-1}) + x(1 + \gamma + \gamma^2 + ... + \gamma^{n-1}) + \\ &\quad \lambda_1(\gamma + 2\gamma^2 + 3\gamma^3 ... + (n-1)\gamma^{n-1}) + \gamma^n R_n \end{aligned}$$

where $R_n = E_{Z_1,...,Z_2n}[V(Z_1 + Z_3 .. + Z_{2n-1} + x, Z_2 + Z_4 + .. + Z_{2n}]$ is the remainder term and $\gamma^n R_n \to 0$. This is because: by Jensen's inequality, we have $E_{Z_1,...,Z_2n}[V(Z_1 + Z_3 .. + Z_{2n-1} + x, Z_2 + Z_4 + .. + Z_{2n}] \leq V(n\lambda_1 + x, n\lambda_2)$, by Corollary 3, we have $V(n\lambda_1 + x, n\lambda_2) \leq \min\{n\lambda_1 + x, n\lambda_2\} + C$, where C is some constant. Therefore by taking $n \to \infty$, we have:

$$V(x,y) \leq \frac{1}{(1-\gamma)^2}(\lambda_1\gamma + (1-\gamma)(\lambda + x)).$$

Similarly, if we recursively apply the relationship that

$$\begin{aligned} V(x,y) &= \lambda + \min\{\gamma\mathbb{E}[V(Z_1+x,Z_2)]+x, \gamma\mathbb{E}[V(Z_1,Z_2+y)]+y\} \\ &\leq \lambda + \gamma\mathbb{E}_{Z_1,Z_2}[V(Z_1,Z_2+y)]+y, \end{aligned}$$

we have that

$$V(x,y) \leq \frac{1}{(1-\gamma)^2}(\lambda_2\gamma + (1-\gamma)(\lambda+y)).$$

Combining these two parts, the result holds. $\square$

Chapter 3

State-Independent Policies

In Chapter 2, we model our problems as an MDP (2.1) and derive some properties of the optimal policies. One feature of the optimal control of (2.1) is that it is state dependent, that is, at each moment, the service provider needs to know the exact queue length of each queue to make his decision. However, in practice, a state-independent policy might be desirable. There are two reasons for this. First, it might be impractical for the service provider to know exactly how many people are waiting in each queue at each moment. Consider the airport shuttle example we gave earlier, when dispatching a shuttle to one location, it is hard to know how many people are waiting there (it is hard and costly to employ a monitoring system for the queue length, moreover, even with a monitoring system, it is hard to know the exact queue length since people are not always standing in a fixed waiting area or may leave during the waiting period, etc.). Second, in real operations, it is very desirable to have a fixed service schedule for both the customers and the service provider. A fixed service schedule not only simplifies the service provider's task, but also relaxes the customers by informing them the next service time. In this section, we propose a state-independent policy for this problem. We first study the optimal structure of such a policy and then show that it performs quite well in test problems.

3.1 Optimal structures of the state-independent policy

In this section, we describe the properties that an optimal state-independent policy should have. In the following discussion, without loss of generality, we assume $\lambda_1 \leq \lambda_2$. We show that the optimal state-independent policy must have the following structure.

Proposition 1. *The optimal state-independent policy should consist of cycles of actions, in each cycle, Q_1 is served once followed by serving Q_2 $k \geq 1$ times.*

In the rest of this section, we are going to prove this proposition.

First, we argue that the optimal state-independent policy must consist of cycles each with a certain service schedule. This is practically meaningful since a cyclic policy is easy to implement, and a schedule based on such a policy is easy to explain, both to the service providers and the customers. To show that this must be the case, note that in a state-independent policies, our decision can only depend on the previous decisions and the distribution of the arrivals (but not the realization of the arrivals). In particular, consider the following specific scenario: Q_1 is served in time period k and Q_2 is served in time period $k+1$. First, note that this scenario must happen for infinite number of k's, otherwise, one queue will be unserved at all from a certain point which is obviously not optimal. The key observation here is that after such a service schedule, the state at the end of time period $k+1$ (or equivalently at the beginning of time period $k+2$) is independent of k. Indeed, at the end of time period $k+1$, the number of people waiting in Q_1 will be $Z_1^k + Z_1^{k+1}$ and the number of people waiting in Q_2 will be Z_2^{k+1}, where Z_1^k's are i.i.d Poisson random variables with mean λ_1 and Z_2^k's are i.i.d. Poisson random variables with mean λ_2. Therefore, the knowledge at the end of time period $k+1$ is the same, regardless of what k is, and one should take the same action (otherwise, the relative better action would be preferred). Therefore, the service schedule between two of such scenarios should be the same, thus a cycling policy has to be used.

Next, we argue that in each cycle, the service schedule must exactly consist of serving queue Q_1 k_1 times followed by serving Q_2 k_2 times (i.e., it cannot happen that a cycle consists the serving sequence of, e.g., $(Q_1, Q_1, Q_2, Q_1, Q_2)$). The reason for this is the same as that in the last paragraph. In particular, we should have the same service schedule between any two such scenarios: Q_1 is served in time period k and Q_2 is served in time period $k+1$. Therefore, the sequence within a cycle only be serving Q_1 k_1 times followed by Q_2 k_2 times for some k_1 and k_2.[1]

Finally, we argue that the optimal service schedule must consist of cycles which alternate between serving Q_1 once, then serving Q_2 k times for some k. To show this, we first show that if both k_1 and k_2 are greater than 1. Then we can find another policy that performs better than the current one. Then we show that if one of k_1 and k_2 has to be one, it must be k_1 (recall that we assume $\lambda_1 \leq \lambda_2$). To show that at optimal, it can't happen that both k_1 and k_2 are greater than 1, we compare the following two cycles:

1. Serve Q_1 k_1 times followed by serving Q_2 k_2 times.
2. Serve Q_1 $k_1 - 1$ times, then serve Q_2, then serve Q_1, then serve Q_2 $k_2 - 1$ times.

First note that both cycles cover $k_1 + k_2$ time periods so we only need to compare the costs within the cycle. Note that the first $k_1 - 1$ actions are the same in both of the cycles, therefore the costs are the same. Now we discuss the cost in the subsequent time periods in both cycles as follows:

- At time period k_1, the cost for the first cycle is the waiting time of the second queue, which is the sum of arrivals in the second queue during the first $k_1 - 1$ periods (which is therefore a Poisson distribution with parameter $(k_1 - 1)\lambda_2$), and the cost for the second cycle is the waiting time of the first queue, which is simply the arrival during period $k_1 - 1$ (which is a Poisson distribution with parameter λ_1).
- At time period $k_1 + 1$, the cost for the first cycle is the waiting time of the first queue, which is a Poisson random variable with parameter λ_1, while the cost for the second cycle is the waiting time of the second queue, which is a Poisson random variable with parameter λ_2.

[1] Because we are looking at long term cost, it doesn't matter how one defines a cycle, i.e., one can either view a cycle as first serving Q_1 a number times followed by serving Q_2 a number of times, or one can view a cycle as first serving Q_2 a number of times followed by serving Q_1 a number of times. We treat them the same. And for the simplicity for the discussion, we assume that Q_1 is first served in a cycle.

- At time after $k_1 + 1$, the cost for Q_2 are the same for both queues, however, obviously the cost for Q_1 in the second cycle is smaller than that in the first cycle since Q_1 is last served in period $k_1 + 1$ rather than k_1.

To summarize, the total costs from time period k_1 and $k_1 + 1$ is $(k_1 - 1)\lambda_2 + \lambda_1\gamma$ in the first cycle and it is $\lambda_1 + \lambda_2\gamma$ in the second cycle. Since $\lambda_1 \leq \lambda_2$ and $\gamma < 1$, the second one is always smaller. Also the total costs from the period after $k_1 + 1$ is also smaller in the second cycle as discussed above. Therefore, the second cycle always has smaller cost than the first one, and thus it is not optimal to have both k_1 and k_2 to be greater than 1.

Lastly, we show that if one of k_1 and k_2 is 1, then it must be k_1. This is intuitive: since the arrival rate λ_1 for Q_1 is smaller, one should expect to serve Q_1 relatively less. To qualitatively illustrate it, one can compare the costs of the following two cycles:

1. Serve Q_1 k times followed by serve Q_2 once
2. Serve Q_1 once followed by Q_2 once, Q_1 $k - 2$ times and then Q_2 once

The cost in the first cycle is $\lambda_2 \sum_{i=0}^{k}(i+1)\gamma^i + \lambda_1\gamma^{k+1}$ and the cost in the second cycle is $\lambda_2 + \lambda_1\gamma + \lambda_1 \sum_{i=2}^{k}(i-1)\gamma^i$. It is easy to see that when $\lambda_1 \leq \lambda_2$ and $\gamma < 1$, the cost in the second cycle is smaller, and thus Proposition 1 is proved.

3.2 Finding the optimal k

In the previous section, we showed that the optimal state-independent policy should consist of cycles, each of which should serve Q_1 once followed by serving Q_2 k times. In this section, we continue to study what is the optimal choice of k.

We do this analytically. For each k, we are going to compute the expected discounted cost. We first look at each cycle.

Proposition 2. *Consider a cycle of $(Q_1, Q_2, ..., Q_2)$ with length $k + 1$ (thus k Q_2s). The expected discounted cost in this cycle is*

$$\lambda_2 + \lambda_1 \sum_{i=1}^{k_2} i\gamma^i + \lambda \sum_{i=0}^{k} \gamma^i. \tag{3.1}$$

Proof. We simply consider the cost in each period and add them up. In period 1, the waiting cost consists of two parts, one is due to the waiting of the customers in Q_2 that hasn't been served. Note that the number of people in Q_2 in the beginning of period 1 must be a Poisson distribution with parameter λ_2 (because before this period, Q_2 must be just served by the cycling structure). Therefore, the expected cost is λ_2. There is also a waiting cost for the new arrivals in period 1, which is $\lambda = \frac{\lambda_1+\lambda_2}{2}$ as we have argued in (2.1). Therefore, the cost in period 1 is $\lambda_2 + \lambda$.

Now we consider the ith period for $i > 1$. Note that in period i, the cost is again due to two parts. One is the waiting time from Q_1. For this part, the total number of people in Q_1 at the beginning of time period i follows a Poisson distribution with parameter $(i-1)\lambda_1$, therefore the expected cost for this part is $(i-1)\lambda_1$. There is also waiting cost due to the new arrivals during that period, which is λ. Therefore, the total cost during period i is $(i-1)\lambda_1 + \lambda$.

To sum them up and take into considerations the discount factor, we have the total expected discounted cost in this cycle is

$$\lambda_2 + \lambda_1 \sum_{i=1}^{k} i\gamma^i + \lambda \sum_{i=0}^{k} \gamma^i.$$

□

Now we consider the total costs throughout the time horizon, if a cyclic policy with length $k+1$ is used, each with cost C, then the total discounted cost will be $C/(1-\gamma^{k+1})$. Therefore we have the following theorem.

Theorem 7. *Assuming the initial state is (M, λ_2) where $M > \lambda_2$. Then the total expected cost of using cyclic policy $(Q_1, Q_2, ..., Q_2)$ with length $k+1$ (thus k Q_2s) is*

$$C(k) = \frac{\lambda_2 + \lambda_1 \sum_{i=1}^{k} i\gamma^i + \lambda \sum_{i=0}^{k} \gamma^i}{1-\gamma^{k+1}}. \tag{3.2}$$

Remark. The reason we involve the big M is to make sure that it is better to serve Q_1 in the first period, and the cost of that is the same for the first period as in later. In practice, if one focuses on long run target, then the costs in the first period does not matter too much and the result still approximately holds, otherwise, a choice of the first step is needed. □

Now our task is simply to find k to minimize $C(k)$. The following theorem shows that $C(k)$ is unimodal in k. For the convenience of notation, we denote $r = \frac{\lambda_2}{\lambda_1} > 1$ in the following discussions.

Theorem 8. *There exists a unique k^* such that $C(k^*-1) \geq C(k^*)$ and $C(k^*+1) \geq C(k^*)$.*

Proof. We compare $C(k)$ and $C(k+1)$. Note that the λ part in $C(k)$ are the same for all k (all equal to $\frac{\lambda}{1-\gamma}$). Therefore, it suffices to compare

$$C'(k) = \frac{\lambda_2 + \lambda_1 \sum_{i=1}^{k} i\gamma^i}{1-\gamma^{k+1}}$$

with $C'(k+1)$. Further by factoring out λ_1, we only need to compare

$$\bar{C}(k) = \frac{r + \sum_{i=1}^{k} i\gamma^i}{1-\gamma^{k+1}}$$

with $\bar{C}(k+1)$. We have $\bar{C}(k+1) \geq \bar{C}(k)$ if and only if

$$(r + \sum_{i=1}^{k} i\gamma^i)(1-\gamma^{k+2}) - (r + \sum_{i=1}^{k+1} i\gamma^i)(1-\gamma^{k+1}) \leq 0. \tag{3.3}$$

By expanding and regrouping terms, we get that (3.3) is equivalent as

$$(1-\gamma)r - (k+1) + (1-\gamma)\sum_{i=1}^{k} i\gamma^i + (k+1)\gamma^{k+1} \leq 0,$$

i.e.,

$$r \leq (k+1)\sum_{i=0}^{k}\gamma^i - \sum_{i=1}^{k} i\gamma^i = \sum_{i=0}^{k}(k+1-i)\gamma^i. \tag{3.4}$$

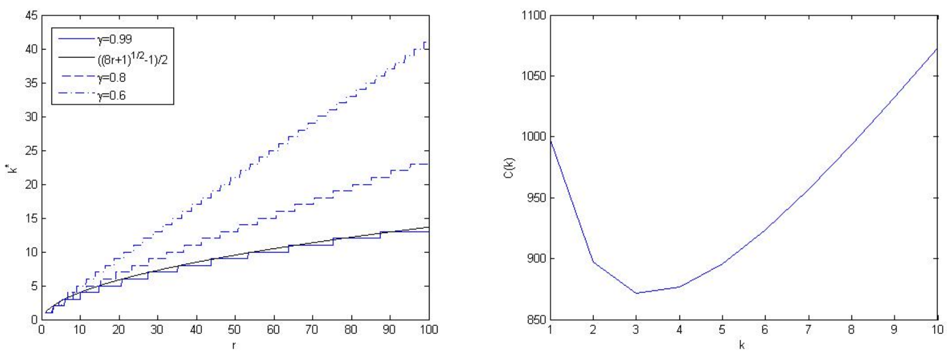
Note that the right hand side of (3.5) is increasing in k. And when $k = 0$, the right hand side is 1 which is less than r, and when k goes to infinity, the right hand also goes to infinity (since it is greater than k). Therefore, there exists a unique integer k^* such that

$$\sum_{i=0}^{k^*}(k^*-i)\gamma^i \leq r < \sum_{i=0}^{k^*+1}(k^*+1-i)\gamma^i. \tag{3.5}$$

Such k^* will satisfy the property stated in the theorem. □

Remark. By Theorem 8, finding the optimal k in the state-independent policy simply becomes to find k that satisfies (3.5). This can be done very fast. Moreover, we can see that k only depends on γ_1 and γ_2 through $r = \frac{\lambda_2}{\lambda_1}$. When r is larger, then k is also larger, and vice versa. This is consistent with our intuition that if the arrival rate of Q_2 is much larger than that of Q_1, we should serve it more frequently.

The next theorem points out the relationship between the optimal k and the discount factor γ and two important limit cases.



(a) Optimal k^*

(b) $C(k)$ with $\lambda_1 = 1$, $\lambda_2 = 9$, $\gamma = 0.8$.

Figure 3.1: Illustrations of k^* and $C(k)$

Theorem 9. *The optimal k is decreasing with the discount factor γ. Furthermore, when $\gamma \to 1$, $k \sim \sqrt{2r} - 1$ and when $\gamma \to 0$, $k \sim r$.*

Proof. The theorem follows immediately from (3.5). □

Theorem 9 is quite instrumental. It directly links the discount factor to the optimal frequency of serving each queue. In particular, when discount factor is small, then we are more focused on the current state, and we will choose the longer queue to serve at each moment. Under the state-independent assumption, this means that if the ratio between the arrival rates is r, then the ratio of the frequency of serving the two queues should be about r (i.e., serve the slow-arrival queue only when it accumulates the same customers as the long queue). On the contrary, when the discount factor goes to 1, we have to be careful about the accumulation effect, that is, the accumulated waiting time for k period for one queue is roughly of order k^2, and we want to serve the slow-arrival queue if k^2 is of order r. Therefore, the ratio of the frequency between serving the two queues should be about $O(\sqrt{r})$. This rule, although simple, may provide important guidance for practitioners when they decide how frequent to serve each queue.

3.3 Numerical experiments

In this section, we perform numerical experiments to study the performance of the state-independent policies. The results are shown in Table 3.1.

In Table 3.1, we list different γ in the first column. In the second column, we choose r from 1 to 9 with their optimal k^*s listed in the third column. We compare the expected costs of three state-independent strategies: (1) serving Q_1 and Q_2 alternately (i.e. $k = 1$); (2) cyclically serving Q_1 once followed by serving Q_2 r times; (3) the optimal state-independent strategy, from the 4th to the 6th columns. We also compute the optimal values (OPT) of the MDP in the 7th column.[2] We compare the costs with the OPT and calculate the gap between them in the last three columns of Table 3.1.

In Table 3.1, we see that the policy k^* indeed performs much better than other choices. In particular, we see that choosing $k = 1$ and $k = r$ usually result in comparable gaps from OPT, while choosing k^* results in about half of the gaps. Furthermore, the gap becomes smaller when r becomes larger, since if one queue has much faster arrival than the other, in either state-independent or dependent policy, one has to serve that queue more frequently, and the discrepancies between these two policies are smaller. Also, the gap becomes larger when γ becomes larger, potentially due to that if the performance is evaluated in a long run, the ability to adjust to current state is more important.

Throughout this chapter, we discuss the structure and performance of the state-independent policy. By showing some numerical results, we conclude this policy is easy

[2] Since the the order of service matters when calculating the numerical data. The OPT value is chosen at $V(M, \lambda_2)$ for sufficient large M. At state $s = (M, \lambda_2)$, Q_1 is served first. Also note that $OPT = V_2^*(\lambda_2)$ by Corollary 1, Q_1 should be served by our optimal policy. We take different γ to test our state-independent policy.

γ	r_s	k^*	$C(1)$	$C(r_s)$	$C(k^*)$	OPT	Gap(1)	Gap(r_s)	Gap(k^*)
0.6	1	1	5.00	5.00	5.00	4.62	8.29%	8.29%	8.29%
0.6	2	1	7.81	7.98	7.81	7.34	6.46%	8.80%	6.46%
0.6	3	2	10.63	10.71	10.51	9.93	6.98%	7.81%	5.82%
0.6	4	2	13.44	13.28	13/04	12.45	7.95%	6.71%	4.72%
0.6	5	3	16.25	15.76	15.51	14.91	8.96%	5.68%	3.97%
0.6	6	3	19.06	18.17	17.90	17.35	9.87%	4.72%	3.19%
0.6	7	4	21.88	20.53	20.28	19.75	10.72%	3.90%	2.68%
0.6	8	4	24.69	22.85	22.62	22.14	11.49%	3.23%	2.15%
0.6	9	4	27.50	25.15	24.95	24.51	12.20%	2.63%	1.82%
0.7	1	1	6.67	6.67	6.67	6.03	10.49%	10.49%	10.49%
0.7	2	1	10.29	10.60	10.29	9.49	8.50%	11.73%	8.50%
0.7	3	2	13.92	14.18	13.79	12.77	9.02%	11.04%	7.99%
0.7	4	2	17.54	17.55	16.98	15.92	10.25%	10.27%	6.67%
0.7	5	3	21.18	20.78	20.14	19.01	11.43%	9.33%	6.00%
0.7	6	3	24.80	23.89	23.13	22.03	12.57%	8.42%	4.96%
0.7	7	3	28.43	26.91	26.11	25.03	13.61%	7.52%	4.34%
0.7	8	4	32.06	29.85	29.03	27.98	14.59%	6.71%	3.75%
0.7	9	4	35.69	32.74	31.90	30.90	15.50%	5.95%	3.23%
0.8	1	1	10.00	10.00	10.00	8.85	13.04%	13.04%	13.04%
0.8	2	1	15.28	15.86	15.28	13.79	10.82%	15.05%	10.82%
0.8	3	2	20.56	21.21	20.41	18.47	11.28%	14.80%	10.49%
0.8	4	2	25.83	26.26	24.96	22.92	12.70%	14.58%	8.89%
0.8	5	2	31.11	31.12	29.51	27.27	14.06%	14.08%	8.18%
0.8	6	3	36.39	35.80	33.79	31.53	15.39%	13.54%	7.14%
0.8	7	3	41.67	40.35	37.98	35.72	16.65%	12.95%	6.33%
0.8	8	3	46.94	44.76	42.17	39.85	17.80%	12.33%	5.83%
0.8	9	4	52.22	49.07	46.20	43.93	18.86%	11.68%	5.16%
0.9	1	1	20.00	20.00	20.00	17.23	16.02%	16.02%	16.02%
0.9	2	1	30.26	31.68	30.26	26.67	13.45%	18.76%	13.45%
0.9	3	2	40.53	42.41	40.37	35.60	13.83%	19.12%	13.39%
0.9	4	2	50.79	52.67	49.06	44.06	15.27%	19.54%	11.34%
0.9	5	2	61.05	62.62	57.75	52.26	16.82%	19.82%	10.50%
0.9	6	3	71.32	73.32	66.13	60.26	18.35%	20.20%	9.75%
0.9	7	3	81.58	81.82	74.04	68.12	19.76%	20.12%	8.69%
0.9	8	3	91.84	91.14	81.95	75.87	21.05%	20.13%	8.01%
0.9	9	3	102.1	100.3	89.86	83.49	22.30%	20.11%	7.63%
0.99	1	1	200.0	200.0	200.0	167.9	19.14%	19.14%	19.14%
0.99	2	1	300.3	316.7	300.3	258.6	16.11%	22.46%	16.11%
0.99	3	2	400.5	424.9	400.3	344.2	16.35%	23.43%	16.30%
0.99	4	2	500.8	529.6	484.0	425.3	17.74%	24.52%	13.80%
0.99	5	2	601.0	632.5	567.7	503.0	19.49%	25.75%	12.86%
0.99	6	3	701.3	734.3	651.0	578.8	21.16%	26.87%	12.48%
0.99	7	3	801.5	835.3	726.4	651.1	23.11%	28.30%	11.57%
0.99	8	3	901.8	935.8	801.8	726.9	24.05%	28.73%	10.30%
0.99	9	3	1002	1035	877.1	799.2	25.37%	29.60%	9.75%

Table 3.1: Performance of different policies

to implement with good performance. In the next chapter, we use the similar analysis method to deal with heterogeneous service time and multi-queues.

Chapter 4

Heterogeneous service time

In Chapter 2 and 3, we assume that the service times for both queues are the same. However, in practice, different queues might need different amount of time to serve. For instance, in the shuttle bus example mentioned earlier, the distances of each pickup locations might be different, which will result in different serving times. In this section, we extend our previous model to accommodate such heterogeneity and derive solutions to this case.

First we write down the MDP formula for this case. In the following, we still use λ_1 to denote the arrival rate of the first queue (Q_1) and λ_2 to denote the arrival rate of the second queue (Q_2). Furthermore, we assume that it takes q_1 units of time to serve Q_1 and q_2 units of time to serve Q_2 (for convenience, we assume q_1 and q_2 are integers in our discussion). Same as in Chapter 2, we use $V(x, y)$ to denote the optimal expected waiting time onward when the state is in (x, y). The Bellman equation can then be written as

$$\begin{aligned} V(x,y) = \min \Bigg\{ & y \sum_{i=0}^{q_1-1} \gamma^i + \lambda \sum_{i=0}^{q_1-1} (1+2i)\gamma^i + \gamma^{q_1} \mathbb{E}[V(Z_1^{q_1}, Z_2^{q_1} + y)], \\ & x \sum_{i=0}^{q_2-1} \gamma^i + \lambda \sum_{i=0}^{q_2-1} (1+2i)\gamma^i + \gamma^{q_2} \mathbb{E}[V(Z_1^{q_2} + x, Z_2^{q_2})] \Bigg\} \end{aligned} \tag{4.1}$$

where $\lambda = \frac{\lambda_1+\lambda_2}{2}$. $Z_i^{q_j}$'s are Poisson random variables with rates $\lambda_i q_j$ and γ is the discount factor. In (4.1), the first term in the braces is the total expected waiting time onward if one chooses to serve Q_1. In particular, the first term $y \sum_{i=0}^{q_1-1} \gamma^i$ is the waiting time of the customers in Q_2 that arrived prior to the current time period, the second term $\lambda \sum_{i=0}^{q_1-1}(1+2i)\gamma^i$ is the waiting time of the customers that arrive during the service of Q_1 and the last term is the cost after the first q_1 periods. Similar explanations apply for the second term in the braces. And the recursion always chooses the smaller term to minimize the value function for state (x, y).

First, one can easily verify that under this model, Theorem 1 and 2 still hold, that is, the value function in (4.1) is increasing and concave, and it is the intersection of two surfaces. The proof follows exactly from the proof of Theorem 1 and 2 and is thus omitted. In the following discussion, we set $\lambda_1 = 1$ without loss of generality and we denote $r = \lambda_2/\lambda_1 > 1$. In Figure 4.1, we plot some examples of the value function with $\gamma = 0.99$.

In the remain of this section, we will discuss the state-independent policy as we did in Chapter 3. We have the following theorem.

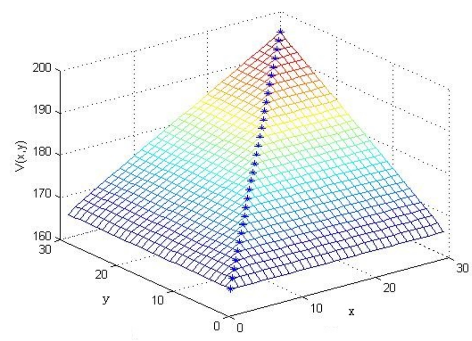
Theorem 10. *The optimal state-independent policy for $q_2 = 1$ is cyclic after certain number periods. In each cycle, Q_1 is served once followed by serving Q_2 $k \geq 1$ times.*

Proof. The proof has a similar structure as in Chapter 3. The cyclic structure has the exactly same proof. To show that $k_1 = 1$, we again compare two cycles:

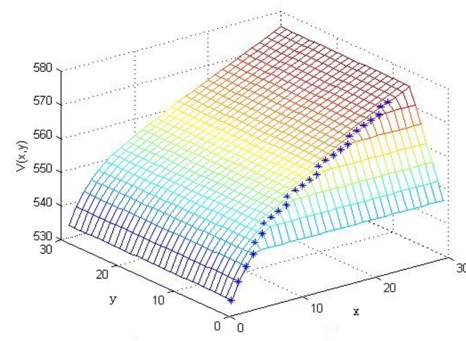
1. Serve Q_1 k_1 times followed by serving Q_2 k_2 times.

2. Serve Q_1 $k_1 - 1$ times, then serve Q_2, then serve Q_1, then serve Q_2 $k_2 - 1$ times.

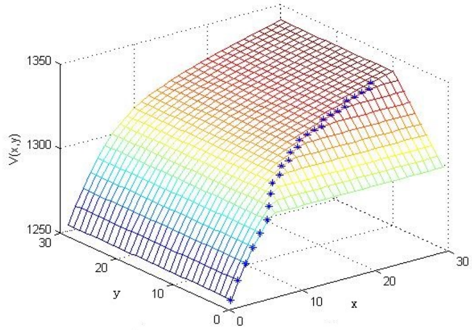
Note that the difference of the two cycles starts from the end of the $(k_1 - 1)$th Q_1 service period and ends at the beginning of the second Q_2 service period. Denote the total expected waiting time in these two service time periods of each cycle as C_1 and



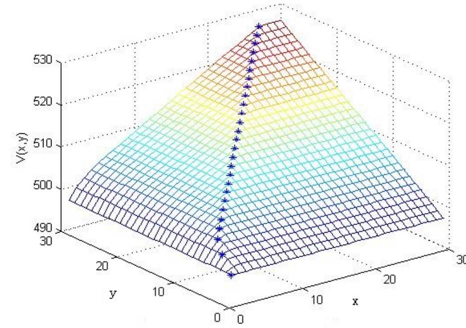
(a) $r = 1$, $q_1 = 1$, $q_2 = 1$



(b) $r = 1$, $q_1 = 5$, $q_2 = 1$



(c) $r = 5$, $q_1 = 5$, $q_2 = 1$



(d) $r = 5$, $q_1 = 1$, $q_2 = 1$

Figure 4.1: Numerical results of $V(x, y)$ with heterogeneous service time

C_2 respectively. We have:

$$\begin{aligned} C_1 &= \lambda_2(k_1-1)q_1\sum_{i=0}^{q_1-1}\gamma^i + \lambda_2\sum_{i=0}^{q_1-1} i\gamma^i + \lambda_2\gamma^{q_1}\sum_{i=0}^{q_2-1} i\gamma^i + \lambda_1\sum_{i=0}^{q_1+q_2-1} i\gamma^i + \\ &\quad \lambda_1(q_1+q_2)\gamma^{q_1+q_2}\sum_{i=0}^{(k_2-1)q_2-1}\gamma^i \end{aligned}$$

and

$$\begin{aligned} C_2 &= \lambda_1 q_1\sum_{i=0}^{q_2-1}\gamma^i + \lambda_1\sum_{i=0}^{q_2-1} i\gamma^i + \lambda_1\gamma^{q_2}\sum_{i=0}^{q_1-1} i\gamma^i + \lambda_2\sum_{i=0}^{q_1+q_2-1} i\gamma^i + \\ &\quad \lambda_1 q_1\gamma^{q_1+q_2}\sum_{i=0}^{(k_2-1)q_2-1}\gamma^i. \end{aligned}$$

The first term of C_1 is the waiting time of the unserved people in Q_2 from the end of the (k_1-1)th Q_1 service period to the beginning of the first Q_2 service period. The second and third terms are the waiting time of the new arrivals in Q_2 during the last Q_1 service period and the first Q_2 service period respectively. The forth term is the waiting time of the new arrivals in Q_1 during the last Q_1 and the first Q_2 service periods. And the last term in C_1 is the cost of the unserved people in Q_1 starts from the beginning of the second Q_2 service period to the end of the cycle. A similar explanation can be applied to C_2. Their difference is:

$$C_1 - C_2 = (\lambda_2(k_1-1)q_1 + \lambda_1 q_2\gamma^{q_2})\sum_{i=0}^{q_1-1}\gamma^i - (\lambda_1 q_1 + \lambda_2 q_1\gamma^{q_1})\sum_{i=0}^{q_2-1}\gamma^i + \lambda_1 q_2\gamma^{q_1+q_2}\sum_{i=0}^{(k_2-1)q_2-1}\gamma^i \quad (4.2)$$

By taking $q_2 = 1$, we have:

$$\begin{aligned} C_1 - C_2 &= (\lambda_2(k_1-1)q_1 + \lambda_1\gamma)\sum_{i=0}^{q_1-1}\gamma^i - (\lambda_1 q_1 + \lambda_2 q_1\gamma^{q_1}) + \lambda_1\gamma^{q_1+1}\sum_{i=0}^{k_2-2}\gamma^i \\ &\geq (\lambda_2(k_1-1)q_1 + \lambda_1\gamma)\sum_{i=0}^{q_1-1}\gamma^i - (\lambda_1 q_1 + \lambda_2 q_1\gamma^{q_1}) \\ &\geq (\lambda_2(k_1-1)q_1 + \lambda_1\gamma)\sum_{i=0}^{q_1-1}\gamma^i - (\lambda_2 q_1 + \lambda_2 q_1\gamma^{q_1}) \\ &\geq (\lambda_2(k_1-1)q_1)\sum_{i=0}^{q_1-1}\gamma^i - (\lambda_2 q_1 + \lambda_2 q_1\gamma^{q_1}) \\ &\geq 0. \end{aligned}$$

Therefore, C_1 is always no smaller than C_2. The theorem is thus proved. $\square$

Remark. This theorem is intuitively true. From Chapter 3, we know that even when $q_1 = q_2 = 1$, the optimal cyclic policy serves Q_1 once in each cycle. Now if $q_1 > q_2 = 1$, that is q_1 is also more time-consuming to serve, then we should keep Q_1's service frequency as minimal as possible, that is $k_1 = 1$. In fact, in our numerical experiment, we find that the optimal k_1 always equals to 1, even when $q_2 > 1$. (If q_2 is large enough, then the optimal cyclic policy is to serve Q_1 once followed by Q_2 once). We conjecture this to be true, yet we are not able to prove it.

Now we consider the total costs throughout the time horizon, if a cyclic policy with length $q_1 + k$ is used, each with cost C, then the total discounted cost will be $C/(1-\gamma^{q_1+k})$. Therefore we have the following theorem.

Theorem 11. *Assuming the initial state is (M, λ_2) where $M > \lambda_2$ and $q_2 = 1$. Then the total cost of using cyclic policy $(Q_1, Q_2, ..., Q_2)$ with length $k+1$ (thus k Q_2s) is*

$$C(k) = \frac{\lambda \sum_{i=0}^{i=q_1+k-1} \gamma^i + \lambda_1 \sum_{i=0}^{q_1+k-1} i\gamma^i + \lambda_2 \sum_{i=0}^{q_1-1} i\gamma^i + \lambda_2 \sum_{i=0}^{q_1-1} \gamma^i}{1-\gamma^{q_1+k}}. \tag{4.3}$$

The term $\lambda \sum_{i=0}^{i=q_1+k-1} \gamma^i$ is the waiting time of the new arrivals in each unit service time. The term $\lambda_1 \sum_{i=0}^{q_1+k-1} i\gamma^i$ is the waiting time of the people in Q_1 and the term $\lambda_2 \sum_{i=0}^{q_1-1} i\gamma^i$ is the waiting time of the people in Q_2 in this cycle. The last term $\lambda_2 \sum_{i=0}^{q_1-1} \gamma^i$ is the waiting time of the unserved people in Q_2 in this cycle.

To find the optimal k that minimize (4.3), we have the following theorem.

Theorem 12. *$C(k)$ is a unimodal function. That is, there exists a unique k^* such that $C(k^*) \le C(k^*-1)$ and $C(k^*) \le C(k^*+1)$.*

Proof. We have $C(k) \le C(k+1)$ is equivalent to:

$$(1-\gamma^{q_1+k+1})(\lambda_1 \sum_{i=0}^{q_1+k-1} i\gamma^i + \lambda_2 \sum_{i=0}^{q_1-1} (1+i)\gamma^i) \le (1-\gamma^{q_1+k})(\lambda_1 \sum_{i=0}^{q_1+k} i\gamma^i + \lambda_2 \sum_{i=0}^{q_1-1} (1+i)\gamma^i). \tag{4.4}$$

By expanding and regrouping, (4.4) is equivalent to:

$$r \sum_{i=0}^{q_1-1} (1+i)\gamma^i \le \sum_{i=0}^{q_1+k} (q_1+k-i)\gamma^i. \tag{4.5}$$

Similarly, $C(k) \le C(k-1)$ is equivalent to:

$$\sum_{i=0}^{q_1+k-1} (q_1+k-1-i)\gamma^i \le r\sum_{i=0}^{q_1-1}(1+i)\gamma^i. \tag{4.6}$$

Therefore, in order for $C(k) \le C(k+1)$ and $C(k) \le C(k-1)$, one needs to have:

$$\sum_{i=0}^{q_1+k-1} (q_1+k-1-i)\gamma^i \le r\sum_{i=0}^{q_1-1}(1+i)\gamma^i \le \sum_{i=0}^{q_1+k}(q_1+k-i)\gamma^i. \tag{4.7}$$

Note that $\sum_{i=0}^{q_1+k}(q_1+k-i)\gamma^i$ is an increasing function of k. Therefore, there exists a unique k^* that satisfy the property stated in the theorem. $\square$

By Theorem 12, finding the optimal cyclic policy becomes to find the k that satisfy (4.7), which can be done efficiently. The next theorem points out the relationship between the optimal k and the discount factor γ and two important limit cases.

Theorem 13. *The optimal k is decreasing with the discount factor γ. Furthermore, when $\gamma \to 1$, $k \sim \sqrt{rq_1^2 + rq_1} - q_1$ and when $\gamma \to 0$, $k \sim 1$.*

Proof. By taking $\gamma = 0$ in (4.7), it's easy to see that $k \sim 1$ as $\gamma \to 0$. Similarly, take $\gamma = 1$, by (4.7) we have:

$$\frac{\sqrt{4rq_1^2 + 4rq_1 + 1} - (2q_1+1)}{2} \le k \le \frac{\sqrt{4rq_1^2 + 4rq_1 + 1} - (2q_1-1)}{2}. \tag{4.8}$$

Simplify (4.8), we have the desired theorem. $\square$

In the following, we conduct numerical experiments to study the performance of the state-independent policies.

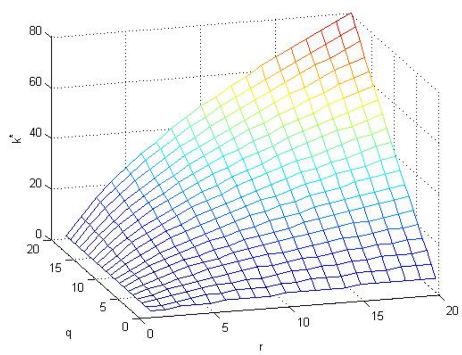
In Table 4.1, we compare the gap between the optimal value and the values of three different policies with $\gamma = 0.99$. The first policy is to cyclically serve Q_1 once followed by Q_2 q times. The second policy is to cyclically serve Q_1 once followed by Q_2 r times. And the third policy is to use our optimal k^* found by (4.7). The first and second columns in the table list different q's and r's in our experiment. The third column lists the optimal k^* in each case. From the 4th to the 7th columns, we present the cost by using these three policies and the optimal value OPT (The OPT value is chosen at $V(M, \lambda_2)$ for sufficient large M). In the last three columns, we compare their gaps with the OPT. We see that the optimal state-independent policy performs well among these

q	r_s	k^*	$C(q)$	$C(r_s)$	$C(k^*)$	OPT	$Gap(q)$	$Gap(r_s)$	$Gap(k^*)$
1	1	1	200.0	200.0	200.0	167.9	19.14%	19.14%	19.14%
1	4	2	500.8	529.6	484.0	425.3	17.74%	24.52%	13.80%
1	7	3	801.5	835.3	726.4	651.1	23.11%	28.30%	11.57%
3	1	1	448.2	398.9	398.9	360.1	24.48%	10.81%	10.81%
3	4	4	901.7	894.6	894.6	795.4	13.36%	12.46%	12.46%
3	7	6	1355.2	1275.1	1272.5	1161.9	16.64%	9.75%	9.52%
5	1	1	694.4	596.6	596.6	551.1	26.00%	8.26%	8.26%
5	4	6	1302.7	1318.8	1298.1	1183.7	10.05%	11.41%	9.66%
5	7	10	1837.8	1910.9	1811.8	1687.7	8.89%	13.22%	7.35%

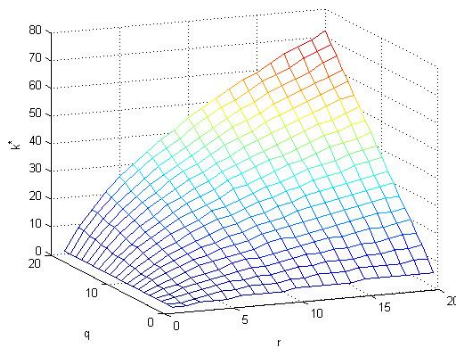
Table 4.1: Performance of different policies with heterogeneous service time

three policies. Furthermore, for the same ratio q, as r increasing, the gap of the optimal policy decreases. Similarly, for the same ratio r, as q increasing, the gap decreases as well.

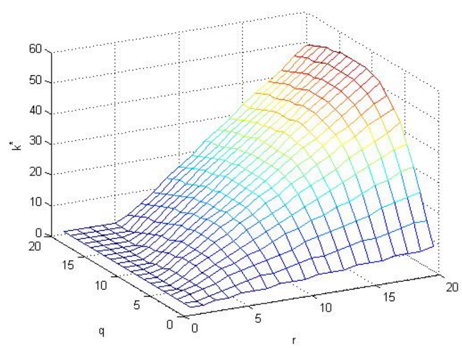
In Figure 4.2, we present the relation of k^* as a function of q and r with different γ's. Note that as $\gamma \to 1$, k^* is increasing in q and r as shown in inequality (4.8). When $\gamma \to 0$, more k^*s equal to 1. These features are consistent with Theorem 13.



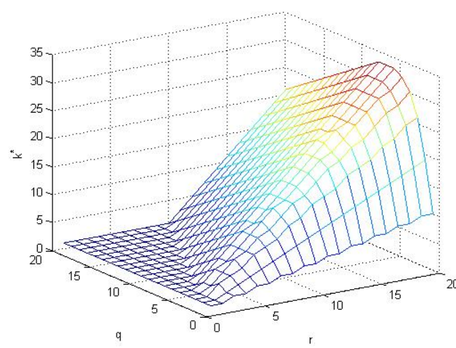
(a) $\gamma = 1$



(b) $\gamma = 0.99$



(c) $\gamma = 0.7$



(d) $\gamma = 0.5$

Figure 4.2: Optimal k^* as a function of q and r

Chapter 5

Conclusion

In this thesis, we consider a special type of two-queue system that has the following features: 1) the customer arrivals in both queues are Poisson processes; 2) there is only one server; 3) once a queue is chosen to be served, all the customers in that queue will be served in a fixed amount of time. The decision in such a system is to decide at each time, which queue should be served and the overall objective is to minimize the total expected delays of the customers.

We first formulate this problem as a Markov Decision Process (MDP). Although the MDP formula is a precise description of the actual problem and the policy obtained therefore is optimal, the optimal policy has the feature of being state-dependent. However, in practice, such a feature might be undesirable due to the limit knowledge of the state of the system and the implementation constraints. To solve this problem, in Chapter 3, we propose a policy that is independent of the current state. Among the state-independent policies, we prove that the optimal one must consist of cycles of a fixed schedule, and each cycle consists of serving the slow-arriving queue once followed by serving the fast-arriving queue multiple (k) times. We then show an explicit formula for the optimal k. In particular, we show that when the discount factor across periods approaches 1, k will be approximately $\sqrt{2r} - 1$, where r is the ratio of arrival rate between the fast-arriving queue and the slow one. This result quantifies the optimal strategy for the service providers of how often should the slow-arriving queue be served, and thus is insightful for practitioners. We implement this strategy and show that it performs well.

We also extend our model to heterogeneous service time. By using the similar analysis method, we obtain the optimal state-independent policy for a simplified model with the service time of Q_2 to be 1. For a more general case, the computation is more complicated and has no such neat form, this is left for our future study.

References

[1] D. Bertsekas. *Dynamic Programming and Optimal Control.* Athena Scientific, 2007.

[2] M. Chaudhry and J. Templeton. *A First Course in Bulk Queues.* Wiley, New York, 1983.

[3] R. Cooper. *Introduction to Queueing Theory, 2nd ed.* Elsevier, North-Holland, New York, 1981.

[4] R. Deb. Optimal dispatching of a finite capacity shuttle. *Management Science*, 24:1362–1372, 1978.

[5] R. Deb and C. Schmidt. Optimal average cost policies for the two-terminal shuttle. *Management Science*, 3(5):662–669, 1987.

[6] M. Dunne and R. Potts. Algorithm for traffic control. *Operations Research*, 12(6):870–881, 1964.

[7] R. Durrett. *Probability: Theory and Examples, Fourth Edition.* Cambridge University Press, 2010.

[8] Laporte G. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operations Research*, 59:345–358, 1992.

[9] D. Gross, J. Shortle, J. Thompson, and C. Harris. *Fundamentals of Queueing Theory, 4th edition.*

[10] E. Ignall and P. Kolesar. Optimal dispatching of an infinite-capacity shuttle: Control at a single terminal. *Operations Research*, 22(5):1008–1024, 1974.

[11] H. Lee and S. Kim. Optimal dispatching of an infinite capacity shuttle with compound poisson arrivals: control at a single terminal. *Computers and Operations Research*, 21(1):67–78, 1994.

[12] H. Lee and M. Srinivasan. The shuttle dispatch problem with compound poisson arrivals: control at two terminals. *Queueing Systems*, 6(2):207–221, 1990.

[13] H. Weiss. The computation of optimal control limits for a queue with batch services. *Management Science*, 25:320–328, 1979.

[14] H. Weiss. Further results on an infinite capacity shuttle with control at a single terminal. *Operations Research*, 29:1212–1217, 1981.